



# Desarrollo y gestión ágil de proyectos Python

## Lección 6: ¿Por qué Scrum?

# ÍNDICE

<b>¿Por qué Scrum?</b>	<b>2</b>
<b>Presentación y objetivos</b>	<b>2</b>
<b>1. Scrum</b>	<b>3</b>
1.1. Visión general	3
1.2. Personas, procedimientos y tecnología	4
1.3. Áreas de responsabilidad y roles habituales	5
<b>2. Pilares de Scrum</b>	<b>7</b>
<b>3. Principios y valores de Scrum</b>	<b>9</b>
1.4. Principios	11
1.5. Valores	12
<b>4. Beneficios de Scrum</b>	<b>13</b>
<b>5. Scrum y Kanban</b>	<b>14</b>
5.1. Kanban	14
5.2. Scrum vs. Kanban	16
<b>6. Puntos clave</b>	<b>21</b>

## ¿Por qué Scrum?

«El mundo está cambiando muy deprisa. No va a ser el grande el que venza al pequeño, sino el rápido al lento.» ~ Rupert Murdoch

### PRESENTACIÓN Y OBJETIVOS

En esta lección sentarás las bases de todo lo aprendido anteriormente para convertirte en desarrollador de proyectos ágiles.



#### Objetivos

En esta lección aprenderás :

- | ***Cuáles son los pilares de Scrum sobre los que debes basar tu trabajo de programación ágil.***
- | ***Los principios y valores sobre los que deberás basar tu trabajo ágil.***
- | ***Las características y beneficios de aplicar Scrum a tu trabajo como desarrollador.***
- | ***Que Scrum y Kanban no son marcos de trabajo rivales como popularmente se cree.***

## 1. SCRUM

La gestión de un proyecto no es ágil por emplear prácticas ágiles, sino por aplicar principios ágiles a la práctica.

### 1.1. Visión general

Hemos hablado a lo largo de la asignatura de cómo la agilidad mejoró el modelo de trabajo habitual basado en ingeniería secuencial y cómo la gestión predictiva hizo ver el desarrollo de software de otra manera llegando a la prácticas ágiles.

El **desarrollo**, basado en estrategia y táctica, evolucionó de una visión de producto completa para convertirse en incremental, al poner en práctica el incremento (iterativo o continuo); convirtiendo, así, el **trabajo** en una sucesión de fases solapadas en lugar de basarse en fases estancas e irreversibles. Con este nuevo modelo de trabajo, el **conocimiento** no está en los procesos, sino que se encuentra en las personas.

#### Visión general: ing. secuencial – ing. concurrente - agilidad

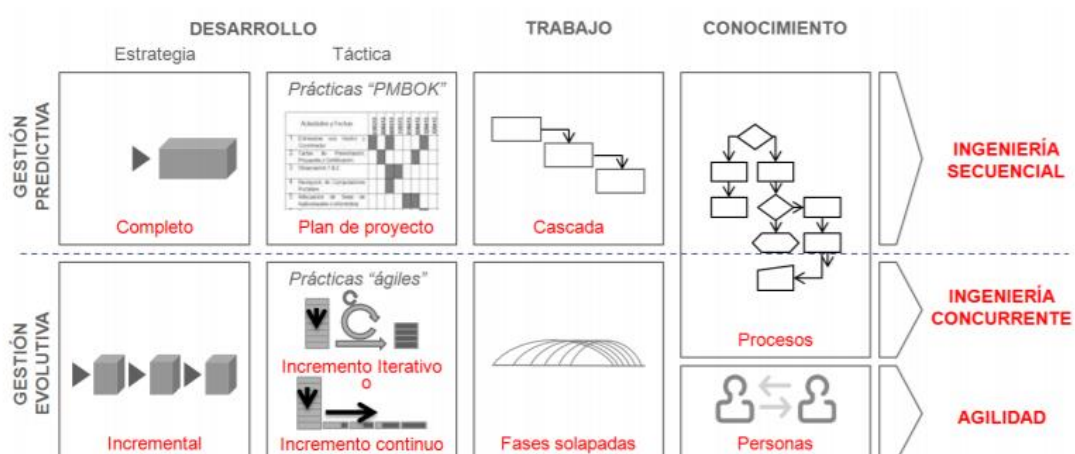
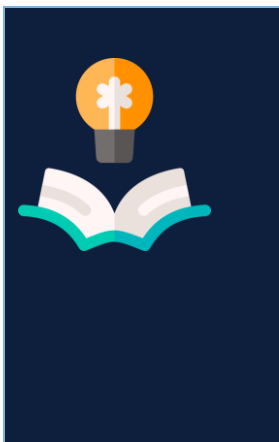


Ilustración 1.1. Visión general

## 1.2. Personas, procedimientos y tecnología

Con este paradigma ágil de desarrollo, un proyecto software ya no se basa en el triángulo personas-procesos-tecnología, sino que los procesos dan paso a los procedimientos, incluyendo tanto proceso como prácticas y las personas son parte fundamental del marco de trabajo, puesto que tienen el conocimiento y son la esencia de la operativa del proceso.



### **Presta atención**

Scrum Manager reconsidera dos vértices del triángulo clásico de los factores de producción: Personas - Procesos y Tecnología. El de procesos y el de personas.



Ilustración 1.2. Personas, procedimientos y tecnología

### 1.3. Áreas de responsabilidad y roles habituales

Hemos aprendido cuáles son los roles habituales de scrum y las funciones de cada uno de ellos.

Ahora vamos a encajar estos roles dentro de una organización.

Evidentemente los equipos scrum no se encuentran aislados, sino que son trabajadores de una organización o empresa. De hecho, en la empresa existirán varios equipos scrum asignados a diferentes proyectos.

Esta organización tendrá un órgano directivo, lo que conocemos como dirección de la empresa y es a nivel técnico donde nos encontramos con los equipos scrum. La figura de scrum master tiene una responsabilidad tanto técnica como avanzada o de organización, ya sus responsabilidades son amplias.

Las responsabilidades de cada uno de los roles habituales en una empresa que aboga por la agilidad para que el sistema funcione pueden resumirse en:

Dirección:

- | Equilibrio sistémico de la empresa.
- | Coherencia del modelo.
- | Medios y formación.

Scrum Master:

- | Configuración de Scrum.
- | Mejora continua.
- | Garantía de funcionamiento de Scrum.

Propietario del producto:

- | Visión del producto.

Equipo:

- | Autoorganización.
- | Tecnología ágil.

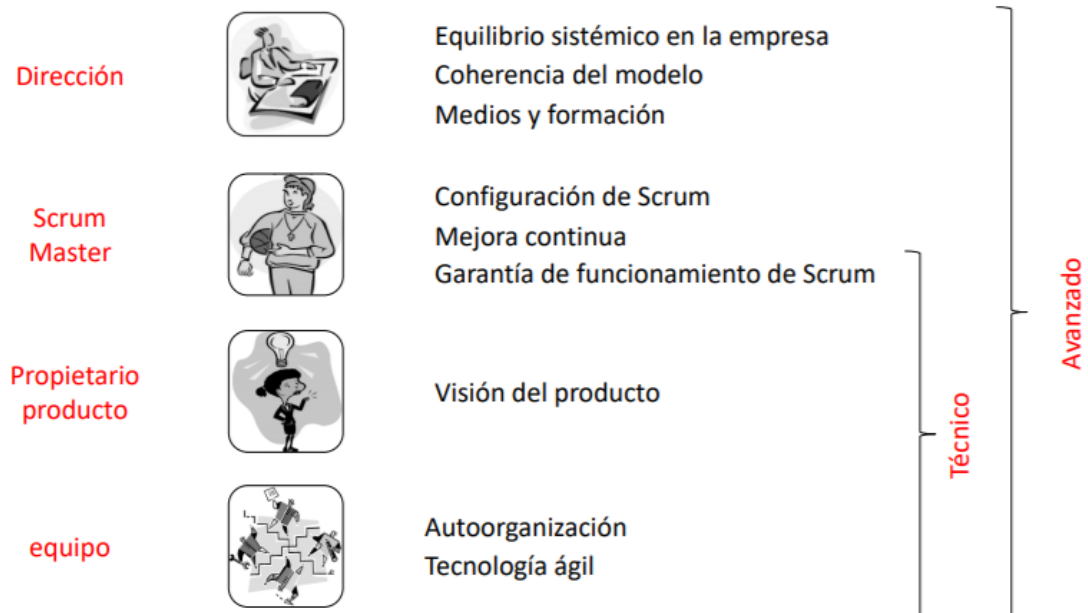


Ilustración 1.3. Áreas de responsabilidad y roles habituales

## 2. PILARES DE SCRUM

Hay tres pilares fundamentales en los que se sustenta Scrum:

1. Transparencia
2. Inspección
3. Adaptación



*Ilustración 2.1. Pilares de Scrum*

### La transparencia

La transparencia implica dar visibilidad a todo lo que está pasando, ya que los aspectos significativos del proceso deben ser visibles para aquellos que son responsables del resultado.

- | La Reunión de Planificación proporciona visibilidad al Equipo Scrum acerca de aquello que va a hacer en el sprint.
- | El Scrum Diario, proporciona visibilidad sobre las tareas diarias, los impedimentos y cómo marcha el trabajo.
- | La Revisión del Sprint ofrece visibilidad sobre los logros, resultados y el progreso.
- | Por último, la Retrospectiva del Sprint contribuye con la inspección y la adaptación del proceso.



## La inspección

Scrum promueve la inspección frecuente de los Artefactos y del progreso para identificar y corregir las variaciones no deseadas.

La inspección tiene lugar durante:

- | La Reunión de Planificación de Sprint,
- | El Scrum Diario,
- | La Revisión del Sprint
- | La Retrospectiva del Sprint.

## La adaptación

La adaptación implica hacer los ajustes en los procesos y artefactos para minimizar la desviación.

En resumen:

- | Transparencia: Deja que se vea qué haces, cómo lo haces y en qué fallas. Es la única forma de mejorar.
- | Inspección: Se debe inspeccionar con frecuencia el producto, la metodología, las herramientas, etc. Mientras antes se encuentre un pequeño problema, antes se evitará un gran problema.
- | Adaptación: El equipo irá minimizando los defectos dando pequeños cambios que ayuden a mejorar el trabajo diario, y por ende, el producto final.

Estos tres pilares, sólo funcionan si se hacen de forma conjunta. De nada nos sirve ser transparentes, si no inspeccionamos nada. De nada sirve la inspección, si después seguimos haciendo lo mismo. Tampoco son útiles los cambios si no los hacemos públicos ni indicamos los efectos positivos y negativos que hemos ido encontrando. El éxito de un equipo Scrum llega cuando éste se sustenta en los pilares que hemos mencionado.

Y es el Scrum Master el que fomenta entre el equipo el cambio de mentalidad para que consiga unos cimientos sólidos donde colocar estos pilares.

### 3. PRINCIPIOS Y VALORES DE SCRUM

Hay algo que no se suele decir en los cursos de scrum: el marco que acabamos de estudiar, las prácticas y herramientas vistas en la primera parte de este manual, no funcionan sin unos principios y valores alineados con ellas.

¿Por qué? Porque las **prácticas** de scrum no son **procesos**. No son pautas que garanticen resultados las ejecute quien las ejecute.

Scrum se asienta sobre el conocimiento tácito de las personas, así como sobre unos valores organizativos.

Las prácticas son sólo las ramas del árbol y, sin unas buenas raíces, no darán frutos. Algo que conviene recordar de vez en cuando para no caer en el error de centrarnos sólo en las herramientas.

Los modelos de gestión facilitan el desarrollo de ciertos principios de trabajo y valores de la organización, sin los cuales trabajaremos en vacío.

Podemos encontrar empresas donde los principios están tan interiorizados que desarrollan su propio modelo de agilidad, con prácticas y técnicas diferentes a las que hemos visto, de otros modelos de agilidad o completamente originales, que dan como resultado productos innovadores y de calidad. Y empresas que, aplicando todas las prácticas del scrum estándar, sólo consiguen empleados alienados, desmotivados, estresados, y resultados mediocres.

Se necesita **flexibilidad** para adaptar lo aprendido a la realidad de cada empresa y proyecto. El objetivo es que la organización sea ágil en su conjunto, capaz de **avanzar en scrum**, en escenarios de trabajo innovadores e inestables.

Para hacer una buena gestión, hay que estar al día de las últimas herramientas y después buscar la manera de ajustarlas al equipo, aprovechando lo que funcione y modificando o desechando lo que no; en lugar de pretender que las personas se adapten a métodos que no les convencen.

Los valores y principios pueden dividirse, aunque sea por pura semántica, en función de si están detrás de las prácticas ágiles o de la cultura de la organización.

Los **principios** son el soporte de las prácticas; los **valores**, de la cultura.

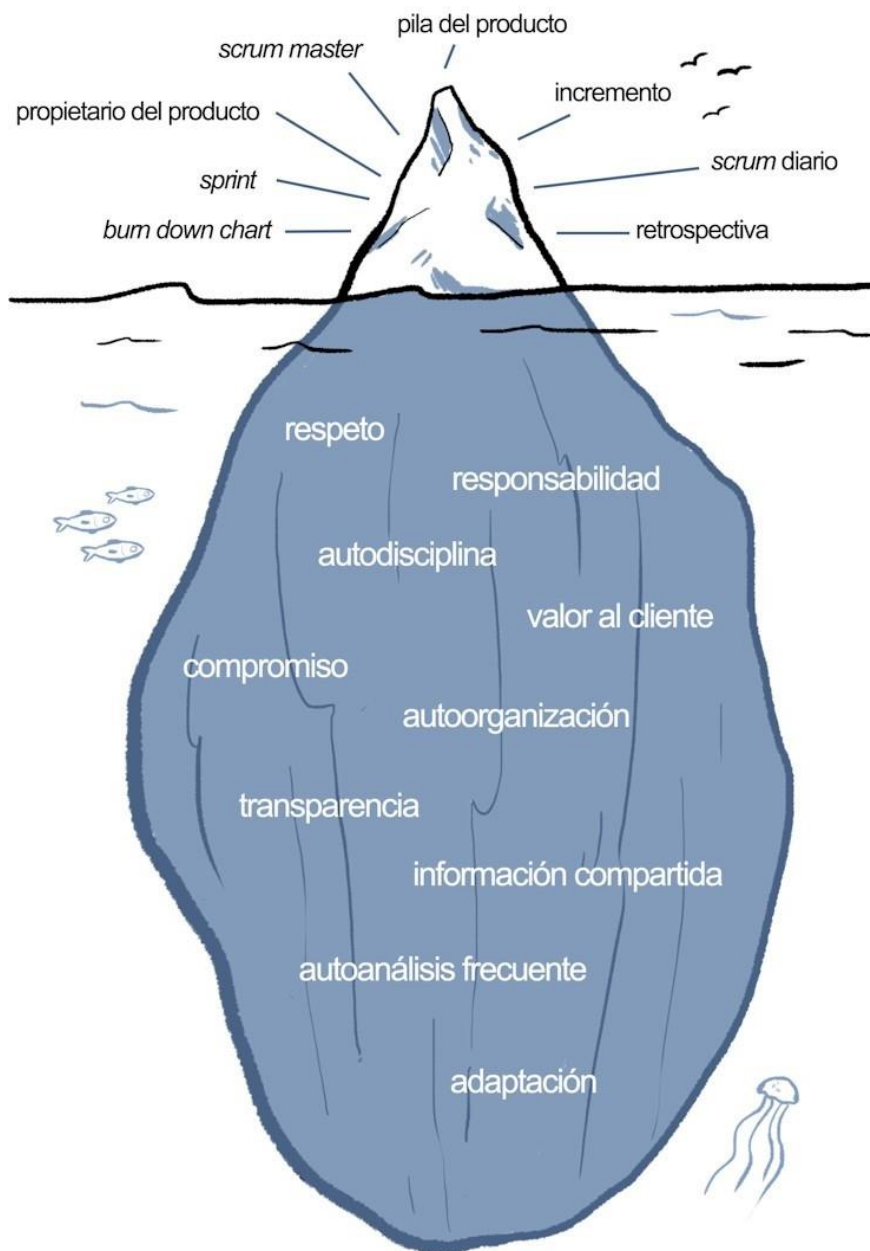


Ilustración 3.1. Principios y valores de Scrum. Fuente: ScrumManager

## 1.4. Principios

La finalidad de artefactos, eventos y otras técnicas ágiles es lograr que el trabajo se base en estos principios.

### **Entrega de valor.**

Entendiendo como tal a la entrega temprana y continua de valor al cliente, para lo que es necesario que éste colabore con el equipo y se comparta y comprenda su visión.

### **Mejora continua.**

En agilidad se reflexiona con frecuencia sobre los métodos de trabajo, cuestionando su efectividad y adaptándolos. El mismo esfuerzo autocrítico se aplica también a la mejora de los productos y servicios que se ofrecen.

### **Desarrollo iterativo e incremental.**

El producto final no se construye conforme a un plan inicial detallado y completo, sino que se arranca desde un «mínimo viable» sobre el que se van añadiendo incrementos.

### **Ritmo de trabajo sostenible.**

Alcanzar un ritmo de trabajo que evite la Ley de Parkinson (el trabajo se expande hasta llenar el tiempo disponible para que se termine) y la presión al descubrir los retrasos demasiado tarde.

### **Atención continua a la excelencia.**

Empleo de técnicas que garanticen la calidad de los productos y servicios y permitan detectar errores con antelación o en el momento de producirse.

### **Operativa visible.**

La información se comparte con claridad para facilitar la colaboración, identificar impedimentos de forma temprana y permitir que todo el equipo conozca el estado del producto y aporte ideas.

**Cadencia y sincronización global.**

Este principio es más relevante cuando se intenta sincronizar a varios equipos que trabajan en productos o servicios relacionados. Se busca predecir la frecuencia de reuniones y fechas de entrega, por ejemplo sincronizando los sprints de los diferentes equipos.

**Personas sobre procesos.**

La inteligencia colectiva del equipo, su conocimiento tácito, es responsable directa de la calidad del producto.

**1.5. Valores**

La cultura de la empresa es la suma de sus características organizacionales y de gobernanza, que pueden acelerar o frenar el desarrollo de la agilidad. En organizaciones cuya cultura se asienta en valores propios de trabajos industriales, basados en procesos, los principios ágiles despliegan resultados de inteligencia colectiva y valor innovador mucho más modestos que en organizaciones con valores que potencian los resultados de trabajar con principios ágiles:

- Asertividad.
- Valoración del talento.
- Claridad.
- Confianza.
- Estructura desjerarquizada.
- Propósito común.

Por último, es necesario el soporte directivo: que la gerencia de la empresa tenga una cultura afín, esté implicada y apoye a las personas con formación y recursos suficientes.

## 4. BENEFICIOS DE SCRUM

Una vez descritas las principales características de Scrum, cabe recapitular y hacer un resumen de sus principales beneficios:

- | Adaptativo, iterativo, rápido, flexible a los cambios y eficaz.
- | Permite la reducción de riesgos, minimizando la cantidad de errores en el desarrollo.
- | Trabajo en equipo.
- | Está enfocado en las personas.
- | Equipo de trabajo autodirigido, autoorganizado y multidisciplinario.
- | Se puede trabajar a distancia ya que cada integrante del equipo sabe lo que debe hacer y lo que está haciendo el resto del equipo.
- | Maximiza el retorno de la inversión.
- | Mayor productividad debido a la eliminación de burocracia.
- | Resultados anticipados (Time to Market) donde el cliente puede obtener resultados importantes y funcionales desde la primera iteración.
- | Entrega de un producto de calidad.
- | Aporta valor de forma rápida.


## 5. SCRUM Y KANBAN

### 5.1. Kanban

Kanban es una técnica de control visual. Es un sistema de señalización para desencadenar acciones.

El término japonés Kanban se puede traducir por tablero o tarjeta de señalización, haciendo referencia a un sistema de señalización empleado en los procesos de producción para coordinar que cada parte se entrega al departamento o sección correspondiente sólo cuando se necesita, reduciendo al mismo tiempo la sobreproducción o almacenamiento innecesarios.

En los años 70 se empezó a popularizar el uso de “Kanban`s” para minimizar el trabajo en proceso (WIP) y reducir los costes asociados al almacenamiento de inventarios.



### *Conceptos*

**Kanban es un sistema de señalización para comunicar información relativa y necesaria en la ejecución o monitorización de un trabajo.**

El desarrollo ágil de software emplea prácticas de gestión visual, por ser las que mejor sirven a los principios de comunicación directa y simplicidad de documentación y gestión.

Conviene resaltar que Kanban no es un modelo ágil, sino que se trata de una herramienta o técnica para comunicar información relativa y necesaria en la ejecución o control de un trabajo. Por tanto, no hay un formato cerrado de tarjetas o tableros Kanban.

Intentar acotar la forma de usar tarjetas con información Kanban, o cómo gestionar las historias o tareas si se emplean tarjetas Kanban, restringe las posibilidades de la herramienta, y va en contra un criterio de gestión flexible para adaptar las técnicas a las características de la organización.



Ilustración 5.1. Kanban de proyecto y Kanban de equipo

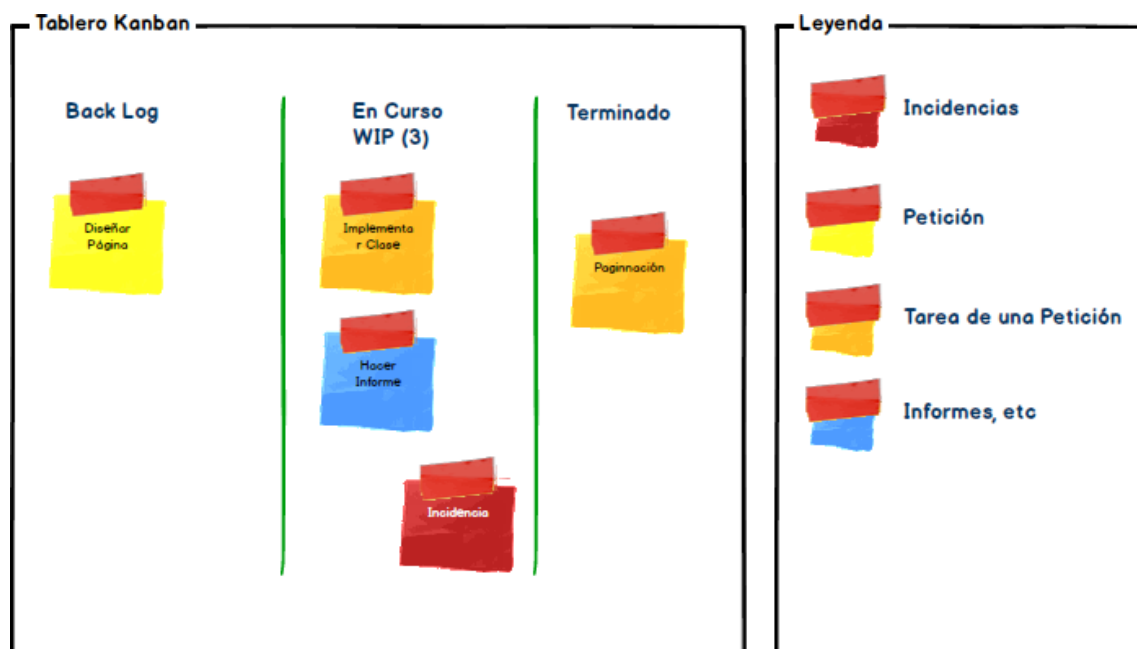
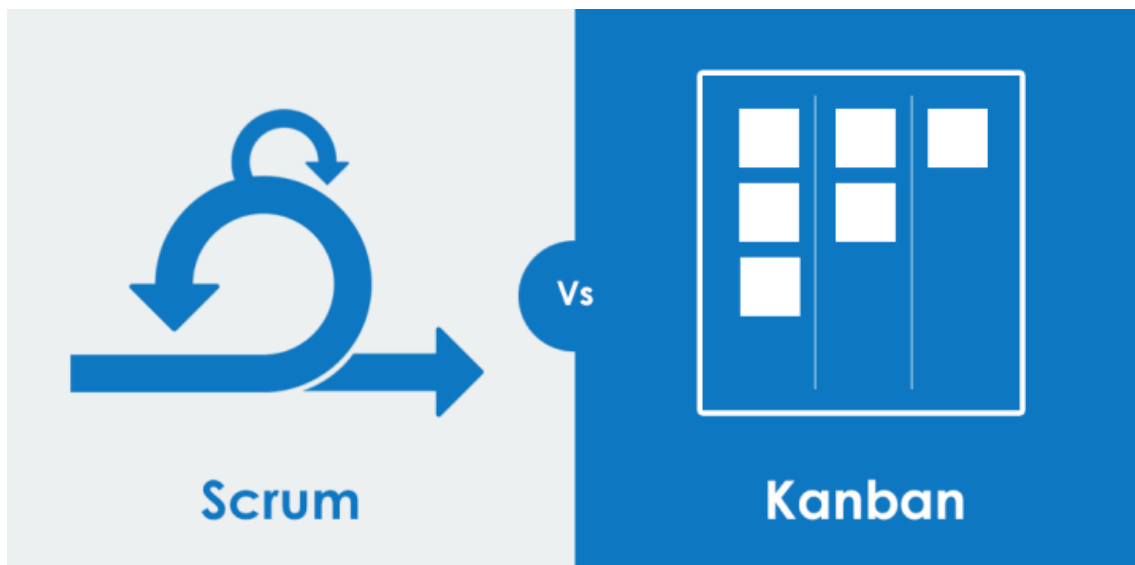


Ilustración 5.2. Tablero Kanban



## 5.2. Scrum vs. Kanban

Una forma de pensamiento habitual es enfrentar Scrum con Kanban, como si uno fuera una alternativa excluyente a la otra, y no sólo no lo son, si no que ambos articulan dos formas distintas de alcanzar un mismo objetivo. Veamos características fundamentales de ambas.



*Ilustración 5.3. Scrum vs. Kanban*

La metodología Scrum se centra en el trabajo por iteración. Esto quiere decir que se basa en la realización de las tareas y cierre de sprints en el tiempo estimado, por lo que la mejora continua será de los procesos y no tanto del proyecto en sí.

La metodología Kanban se enfoca en el flujo de trabajo. Lo que quiere decir que se centra más en el estado de los proyectos y no tanto en las tareas o acciones a llevar a cabo, por este motivo no puede haber más de un número determinado de proyectos en la misma fase de desarrollo.

Tanto en Scrum como Kanban, es necesario disponer de un equipo altamente auto organizado.

Un punto importante que las diferencia es que con una metodología Scrum es necesario disponer de un Scrum Master, que es la figura administrativa del proyecto, es el encargado de asegurar que el resto del equipo (Scrum Team o equipo de desarrollo).

Por el contrario, la metodología Kanban necesita menos supervisión administrativa ya que está más pensado para gestionar el propio flujo de trabajo, dejando a cada integrante del equipo trabajar de forma autónoma.

En cuanto a los tiempos, Scrum habitualmente se planifica por semanas, desarrollando un sprint por semana. Se suele hacer una reunión semanal para poner en común el estado de las acciones y así determinar siguientes pasos o variaciones en los próximos sprints.

Mientras que en Kanban no hay una línea de tiempo predeterminada, sino que el trabajo se analiza y valida con frecuencia para poder avanzar al siguiente estado.

Por ello, suele hacerse una reunión siempre que sea necesario validar o mostrar parte de los avances del proyecto.



### ***Presta atención***

**En un nivel pragmático la diferencia no es “scrum o kanban” sino el avance en sprints con un flujo continuo.**

En un nivel de scrum pragmático la diferencia no es “scrum o kanban” sino avance en sprints o con un flujo continuo.



Son apropiadas las **técnicas de gestión visual kanban** para evitar los cuellos de botella y los tiempos muertos.

Ajustándolas con criterios de flexibilidad a las circunstancias de nuestro trabajo y equipo

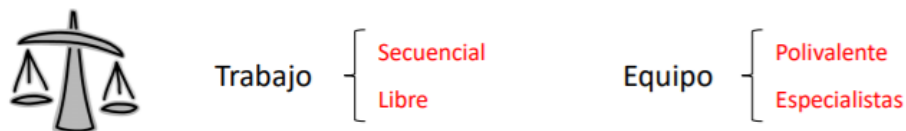


Ilustración 5.4. Scrum y Kanban

## Las 10 diferencias

A modo de resumen vamos a ver 10 diferencias que existen entre Scrum y Kanban:

1. En Scrum existen los roles de Scrum Master, de Product Owner y del equipo, mientras que Kanban no existen roles.
2. En Scrum se trabaja con iteraciones de tiempo fijo, con unos ciclos fijos que se denominan Sprints, mientras que en Kanban tenemos un trabajo continuo y no tenemos esas iteraciones o esos ciclos durante el desarrollo.
3. Scrum limita el WIP (work in process o número de tareas que se pueden tener en paralelo en una de las posiciones del tablero) por iteración, mientras que Kanban limitar ese WIP por el estado del flujo de trabajo.

4. Mientras que Scrum exige equipos multidisciplinarios, para que todos los miembros del equipo puedan realizar varias tareas y sea todo más más ágil, en Kanban se permiten los equipos formados por especialistas. En este caso podemos tener algún problema a la hora de gestionar esos equipos, pero existen una serie de normas o de prácticas para llevar a cabo para solucionarlo.
5. En Scrum no se permiten cambiar las tareas del Sprint, es decir, una vez que la tarea asignada al mismo no puede ser movida, en todo caso lo que se permite es modificar la fecha de entrega del Sprint, pero no esa tarea. En Kanban, por el contrario, se puede modificar la tarea hasta que entra en flujo, hasta ese momento podemos modificar la tarea.
6. En Scrum la pila del producto, es decir, el conjunto de tareas que tenemos que realizar durante el Sprint, tiene que tener al menos el tamaño de un Sprint, ya que, lógicamente, no podemos tener menos de un Sprint. En Kanban, al tener un ritmo de trabajo continuo, lo que se hace es ir arrastrando las nuevas tareas por el panel hasta que lleguen a su estado final y finalicen. Cualquier nuevo requisito del cliente será una nueva tarjeta o post-it que se añadirá al flujo de entrada y que seguirá su flujo normal hasta la salida.
7. En Scrum se mide todo lo que sea necesario, se miden historias, es decir, cuánto nos va a llevar realizar cada historia de usuario, se mide cuánto tiempo o esfuerzo nos va a llevar realizar cada una de las tareas y se mide también la velocidad del equipo, que es la cantidad de trabajo que hemos realizado dividido por la cantidad de tiempo. En Kanban, como ya tenemos una cierta habilidad de la metodología, no se miden ni tareas ni velocidad.
8. En Scrum se necesita una pila del producto priorizada, porque como el desarrollo completo lo vamos a dividir en distintos Sprints, la necesitamos para que los primeros Sprints se encarguen de realizar las tareas de mayor prioridad. Con lo que vamos a conseguir llevar valor al cliente de una manera mucho más rápida y las tareas con menor prioridad serán realizando en los Sprints posteriores. Esta priorización

la hará el Product Owner. En Kanban la historia o tarea es arrastrada directamente desde el cliente, por lo cual no necesita esa priorización.

9. En Scrum se tienen una serie de reuniones y se utilizan una serie de gráficos, como el burn-down, en el que podemos ver el avance del proyecto. En Kanban no se considera ni ese tipo de reuniones ni de gráficos.
10. En Scrum los tableros se van a resetear al final de cada Sprint, es decir, conforme vamos finalizando el mismo, el tablero queda vacío y comenzamos de nuevo añadir nueva nuevas historias de usuario, las siguientes en prioridad. En Kanban, vamos a tener un flujo de entrada-salida, conforme las tarjetas van pasando por cada uno de los estados hasta llegar al estado final, cuando llegan a ese estado salen del tablero y se archivan, vamos a tener un flujo continuo.

## **¿Entonces qué framework ágil es mejor?**

A día de hoy no cabe duda que Scrum se presenta como claro favorito, si atendemos únicamente a los números.

Que Scrum es el método preferido en nuestro país, donde de facto se ha convertido en un falso sinónimo de Agile, nadie lo puede discutir. Que Kanban se conoce más como un tablero lleno de post-its de colores, que, como una implementación de Agile completa, es otro hecho destacable, al menos en la capa de gestión de las compañías.

Kanban se basa en el desarrollo y entrega continuos, abordando un pequeño número de tareas de forma fluida y simultánea. Los equipos Kanban utilizan una herramienta de planificación visual, el tablero Kanban (ése que hablábamos con los post-its), que muestra cada proyecto (historia del usuario) en una tarjeta y mueve esas tarjetas a través de columnas que representan etapas progresivas de finalización.

Si tu equipo tiene un flujo continuo de solicitudes de trabajo, el Kanban puede ser el adecuado para gestionarlo. Por ejemplo, muchos equipos de soporte y mantenimiento usan Kanban en vez de Scrum, ya que con este último la tarea de gestión de su trabajo se hace casi imposible.

Scrum también divide tareas complejas en historias de usuarios y las visualiza en un flujo de trabajo (muchas veces en un tablero Kanban). Los equipos de Scrum se comprometen a generar software al final de los intervalos establecidos, Sprints.

Si tu necesidad es entregar valor a los clientes de forma regular, scrum parece mejor camino.

	SCRUM	KANBAN
<b>Cadencia</b>	Sprints regulares de duración limitada ( 2-4 semanas )	Flujo continuo
<b>Entrega de valor</b>	Al final de cada Sprint, con la aprobación del Product Owner	Entrega continua o a discreción del equipo
<b>Roles</b>	Product Owner, Scrum Master, Dev. Team	No hay roles. Algunos equipos incluyen un Agile Coach.
<b>Métricas</b>	Velocidad	Tiempo de ciclado
<b>Filosofía del cambio</b>	No hay cambio durante el Sprint.	El cambio puede ocurrir en cualquier momento.

Teniendo en cuentas todo lo indicado debes escoger la opción que mejor se adapte a las necesidades de tu organización, proyecto y equipo de trabajo. Aunque, sin duda, el marco ágil más extendido hoy día es Scrum.

## 6. PUNTOS CLAVE

- | Asentamiento final de conocimientos de Scrum: visión, principios, valores, componentes, áreas, características y beneficios.
- | Comparativa Scrum y Kanban.

